

# Введение в Программирование

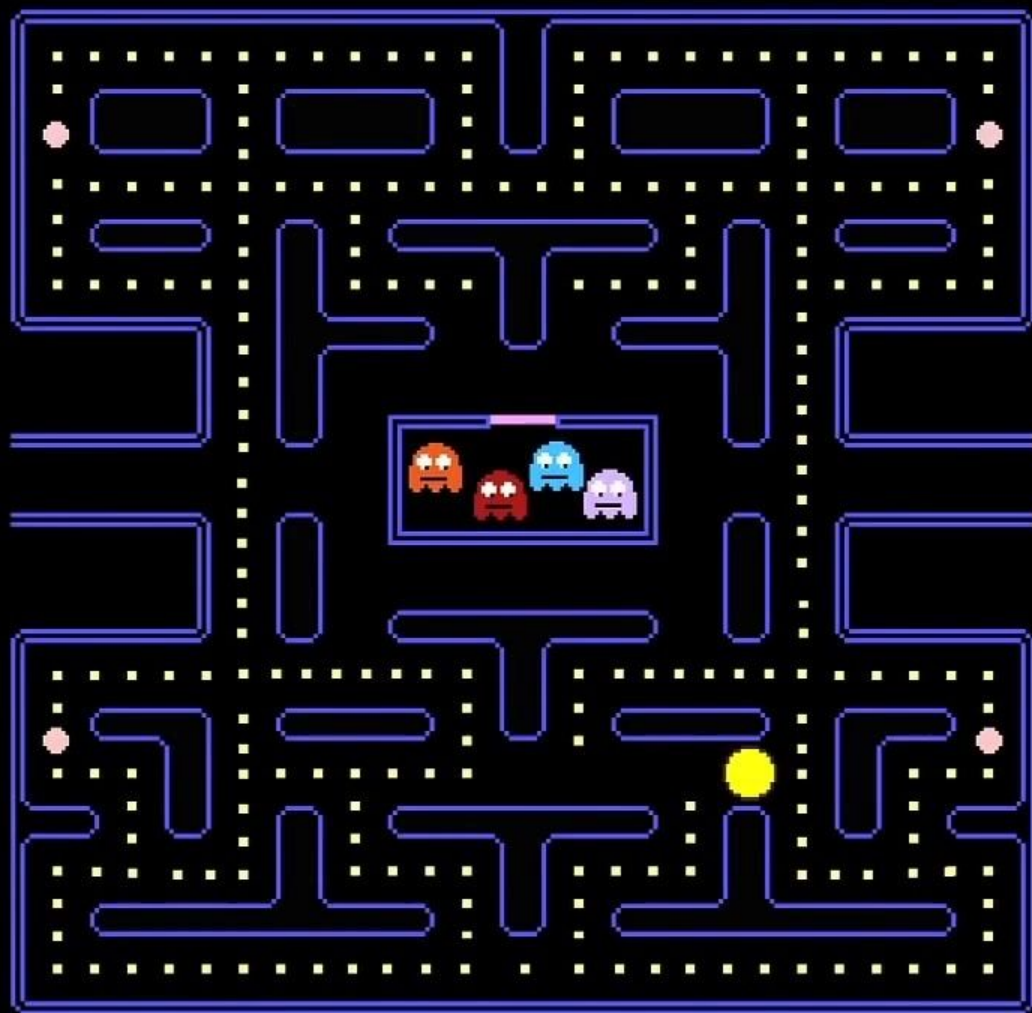


# Сегодня

1. Введение в введение
2. О программировании в целом и web в частности
3. Как сделать сайт? И что для этого надо?
4. Первая программа
5. Основные конструкции языка

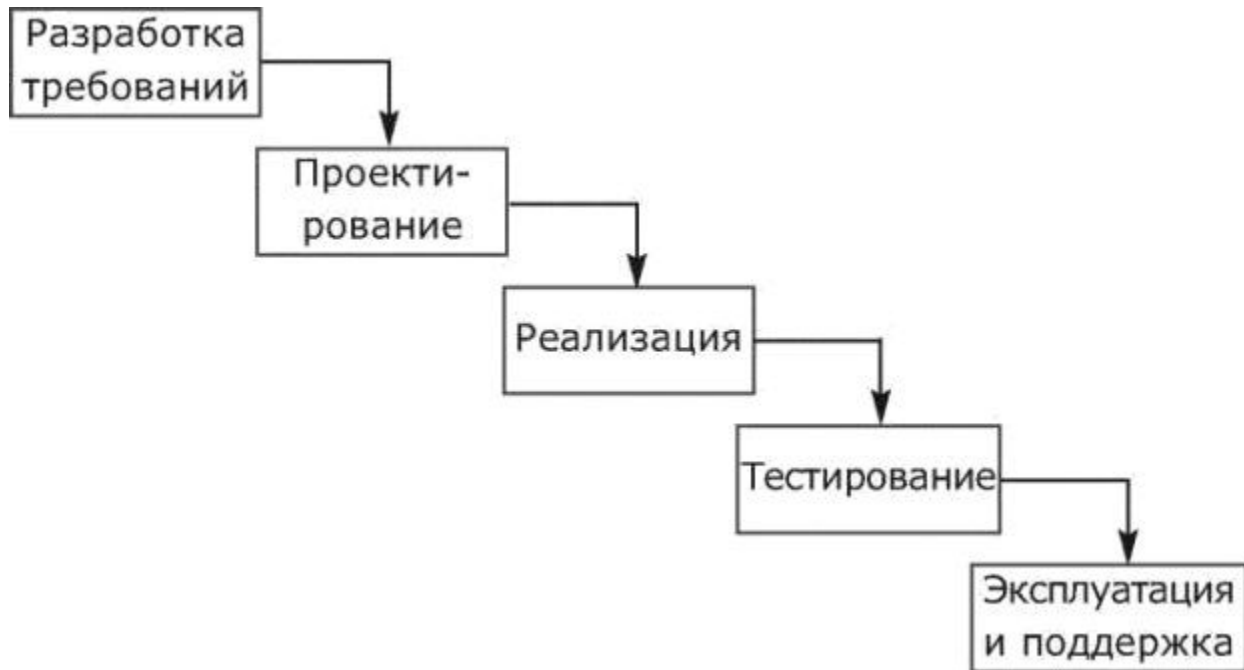
## О чем этот курс?

- О Javascript? - нет
- О программировании? - да!
- О любви к тому, что делаешь? - **да!**

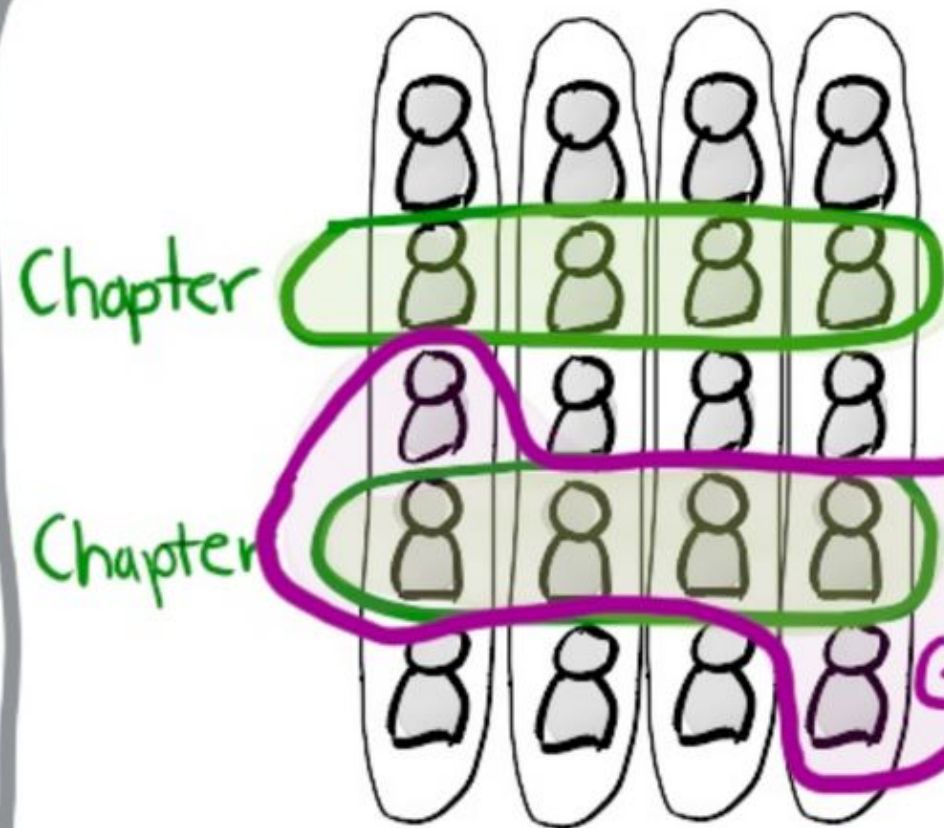


# Как создается программное обеспечение?

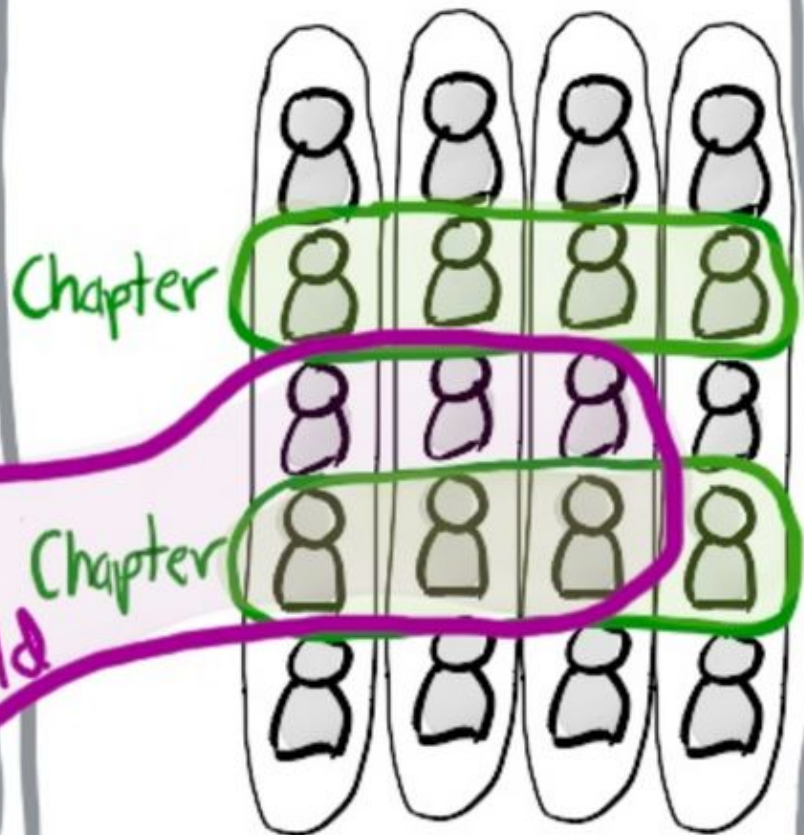




Tribe



Tribe



# Программистов в команде много. Что с того?

- Особые методологии организации рабочего процесса
  - требования постоянно меняются
  - куча творческих людей
  - высокая конкурентность на рынке труда
- Соглашения о стиле кодирования



# Пример кода

```
/** @param {number} foo */  
method(foo) {  
  if (condition(foo)) {  
    try {  
      // Note: this might fail.  
      something();  
    } catch (err) {  
      recover();  
    }  
  }  
}
```

Вернемся в программирование

# Дадим слово Википедии

В программирование входят

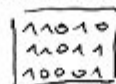
- анализ и постановка задачи,
- проектирование программы,
- построение алгоритмов,
- разработка структур данных,
- написание текстов программ,
- отладка и тестирование программы (испытания программы),
- документирование,
- настройка (конфигурирование),
- доработка и сопровождение.

Source code:  
hello.c



→ COMPILER →

Machine code:



Program (also  
called binary,  
executable ...)

run the  
program

result



Source code:  
hello.py



→ INTERPRETER → result



# Особенности веб-программирования

- Разделение на frontend и backend
- Большая нагрузка
- Много данных
- Сложная инфраструктура



Программист ставит себе на тумбочку перед сном два стакана.  
Один с водой - на случай, если захочет ночью пить. А второй  
пустой - на случай, если не захочет.

Что нужно чтобы сделать сайт?



Блокнот



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Мой Первый Сайт</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Hello World!</title>
```

```
</head>
```

```
<body>
```

```
  <script>
```

```
    alert("Hello, World!");
```

```
  </script>
```

```
</body>
```

```
</html>
```

# Комментарии

**<script>**

*//Однострочный комментарий*

*/\**

*Многострочный комментарий*

*\* /*

**</script>**

# Переменные

```
var name = "Строка"; //строковая переменная  
var digit = 42; //числовая переменная  
var isProgrammer = true; //логическая переменная
```

# Особенности выбора имени переменной

- Должно соответствовать данным, содержащимися внутри
- Нельзя начинать с цифры
- Соглашение: именуем в camelCase
- Регистр символов имеет значение

В JS типов переменных в разы больше, но пока нам нужны только эти 3.

## Условные операторы

```
var isPasswordCorrect = true;  
if (isPasswordCorrect) {  
    alert("Hello!")  
} else {  
    alert("Error!");  
}
```



# Сравнения

```
var a = 5;
```

```
var b = 7;
```

```
alert(a < b);
```

```
alert(a > b);
```

```
alert(a >= b);
```

```
alert(a <= b);
```

```
alert(a == b);
```

```
alert(a != b);
```

# Взаимодействуем с пользователем

```
alert("Hello!"); //Выводим сообщение
prompt("Enter your name: ", ""); //Получаем данные
ПОЛЬЗОВАТЕЛЯ
confirm("Вам больше 18?"); //Получаем данные пользователя (y/n)
var food = prompt("Введите название еды", "");
if (food == "Конфета") {
    alert("Я не ем сладкое");
} else {
    alert("Ом-ном-ном!");
}
```

# Консоль

```
console.log("Hello, world!");
```

## Цикл While

```
while (true) {  
    confirm("Уже приехали?");  
}
```



Перебираем символы в строке

## Версия 1. Создадим счетчик

```
var templateString = "HelloWorld#";  
var counter = 0;  
while (templateString[counter] != "#") {  
    console.log(templateString[counter]);  
    counter = counter + 1;  
}
```

## Вариант 2. Длина строки

```
var templateString = "HelloWorld";  
var counter = 0;  
while (counter < templateString.length - 1) {  
    console.log(templateString[counter] );  
    counter = counter + 1;  
}
```

## Вариант 3. Цикл for

```
var str = "HelloWorld";  
for (var i = 0; i < str.length; i++) {  
    console.log(str[i]);  
}
```



# ФУНКЦИИ

```
function showMessage() {  
    alert("Привет, Мир!");  
}
```

```
showMessage();
```

# Параметры функции

```
function printHello(userName, greetingText) {  
    var message = greetingText + userName;  
    console.log(message);  
}
```

```
printHello("Alex", "Hello ");
```

# Возвращаемые значения

```
function getAlexNameByAge (age) {  
    var name = "";  
    if (age > 18) {  
        name = "Алексей";  
    } else {  
        name = "Алёша";  
    }  
    return name;  
}
```

```
var alexName = getAlexNameByAge(18);  
printHello(alexName, "Hello, ");
```

# Что еще может пригодиться

$*$ ,  $/$ ,  $+$ ,  $-$  - операции

$\%$  - остаток от деления